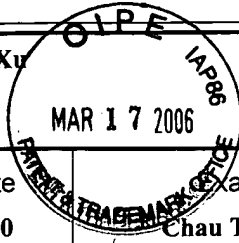


#AF/PW

TRANSMITTAL OF APPEAL BRIEF (Large Entity)

Docket No.  
ITL.0403US

In Re Application Of: Yueheng Xu



Application No.  
09/748,895

Filing Date  
12/27/2000

Examiner  
Chau T. Nguyen

Customer No.  
21906

Group Art Unit  
2176

Confirmation No.  
5241

Invention: Large Character Set Browser

COMMISSIONER FOR PATENTS:

Transmitted herewith in triplicate is the Appeal Brief in this application, with respect to the Notice of Appeal filed on January 19, 2006

The fee for filing this Appeal Brief is: \$500.00

- ☒ A check in the amount of the fee is enclosed.
- ☐ The Director has already been authorized to charge fees in this application to a Deposit Account.
- ☒ The Director is hereby authorized to charge any fees which may be required, or credit any overpayment to Deposit Account No. 20-1504
- ☐ Payment by credit card. Form PTO-2038 is attached.

**WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.**

Signature

Mark J. Rozman, Reg. No. 42,117  
TROP, PRUNER & HU, P.C.  
8554 Katy Freeway, Suite 100  
Houston, TX 77024

Dated: March 15, 2006

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to "Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450" [37 CFR 1.8(a)] on

03/15/2006

(Date)

Signature of Person Mailing Correspondence

Jennifer Juarez

Typed or Printed Name of Person Mailing Correspondence

cc:



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: YUEHENG XU

Serial No.: 09/748,895

Filed: December 27, 2000

For: LARGE CHARACTER SET  
BROWSER

§ Group Art Unit: 2176

§

§

§ Examiner: Chau T. Nguyen

§

§

§ Atty. Dkt. No.: ITL.0403US (P8986)

§

Mail Stop Appeal Brief-Patents  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

**APPEAL BRIEF**

03/17/2006 MBIZUNES 00000034 09748895

01 FC:1402

500.00 OP

Date of Deposit: March 15, 2006

I hereby certify under 37 CFR 1.8(a) that this correspondence is being deposited with the United States Postal Service as **first class mail** with sufficient postage on the date indicated above and is addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

  
Jennifer Juarez



## TABLE OF CONTENTS

I.	REAL PARTY IN INTEREST	3
II.	RELATED APPEALS AND INTERFERENCES	4
III.	STATUS OF CLAIMS	5
IV.	STATUS OF AMENDMENTS	6
V.	SUMMARY OF CLAIMED SUBJECT MATTER	7
VI.	GROUND OF REJECTION TO BE REVIEWED ON APPEAL	10
VII.	ARGUMENT	11
VIII.	CLAIMS APPENDIX	17
IX.	EVIDENCE APPENDIX	None
X.	RELATED PROCEEDINGS	None

**I. REAL PARTY IN INTEREST**

The real party in interest is the assignee Intel Corporation, the assignee of the present application by virtue of the assignment recorded at Reel/Frame 011413/0729.

## **II. RELATED APPEALS AND INTERFERENCES**

There are no related appeals and interferences.

### **III. STATUS OF CLAIMS**

Claims 1-7, 11-17, 21-24 and 26-32 stand rejected. The rejections of all pending claims 1-7, 11-17, 21-24 and 26-32 are being appealed.

#### **IV. STATUS OF AMENDMENTS**

All amendments have been entered. No amendment has been filed subsequent to the final rejection.

## **V. SUMMARY OF CLAIMED SUBJECT MATTER**

At this point, no issue has been raised that would suggest that the words in the claims have any meaning other than their ordinary meanings. Nothing in this section should be taken as an indication that any claim term has a meaning other than its ordinary meaning.

A web browser 12, shown in Figure 1, provides a platform level solution to the need for larger character sets. The web browser 12 may serve as a universal, cross-platform user interface for any web-based application. The browser 12 effectively enables a large character set to support all web-based applications. Specification, p. 4, lns. 1-8.

A source hypertext markup language (HTML) file is encoded in accordance with extension formats as indicated in block 10. In the browser 12, that file is converted into a Unicode format extended to include support of a surrogate mechanism of Unicode. Using the surrogate mechanism of Unicode extends Unicode to a capacity sufficient to recognize a character set of one million unique characters. Since existing Unicode standards use a sixteen bit character length, the sixteen bit value can provide no more than 65,536 characters.

Thus, the extension formats are converted to Unicode with surrogate support as indicated at block 14. Those CJK characters that are already defined in Unicode standard are converted into their sixteen bit Unicode value. The remaining characters that are not defined by the Unicode standard are converted into two sixteen bit values. Specification, p. 4, lns. 9-25.

The browser 12 does the web content parsing, as indicated in block 16, such as HTML parsing. Thereafter, the remaining rendering steps are completed as indicated in block 18. Specification, p. 5, lns. 4-7.

A Chinese web page content is composed of English HTML tags and Chinese text strings encoded according to a known Chinese character set standard. When the web page is read into the browser, all of its content, including English tags and encoded Chinese text strings are first converted into Unicode text strings. This Unicode text string is then parsed by a HTML parser and a Document Object Model is built to represent that abstract data structure of a web page. Then a rendering engine is applied to walk through the Document Object Model to visit and render each element of the page using the formats and fonts associated with the HTML tags.

During the rendering process, the rendering engine will render one Unicode character at a time. For each particular Unicode character, the rendering engine runs a fast check against an availability index matrix of that font file to see if the font for a given Unicode character is



available in a given font file. If the character does not exist in the font file, the rendering engine looks for the next font file to search for a font that is as close to the style as possible.

Specification, p. 7, lns. 5-25.

When a font is found that corresponds to the character, the Unicode of that character is converted back to a national character set encoding, and the value of this encoding is used to index the offset of this character's font bytes in a given font file. Then, the rendering engine fetches those font bytes and generates a glyph image for that character in a given location calculated by a layout engine.

During this process, there are two steps where there is a character set conversion. One step is converting the web page source into Unicode before it gets parsed. The other step is converting each Unicode character in a text string to its character set encoding standard so that its value can be used as an index to fetch font information in a font file. Specification, p. 8, lns. 1-15.

For a web page whose Chinese characters are encoded according to ISO-2022-CN-EXT, where the Chinese characters are arranged in several 95x95 planes, 100,000 Chinese characters may be arranged in twelve planes, each plane having 95 rows (0x20, 0x21, ..., 0x7f) and 95 columns (0x20, 0x21, ..., 0x7f). These planes are the same as these ISO-2022 planes. Each plane has  $9025((0x7f-0x20) \times (0x7f-0x20) = 95 \times 95)$  characters. Twelve planes hold all of the Chinese characters. Sixteen planes hold about 150,000 characters, which is more than all the possible CJK and Vietnamese characters added together. Specification, p. 10, ln. 24 – p. 11, ln. 9. In summary, the font files are divided into planes, each plane having fonts for 95x95 characters in one embodiment. The font for a particular character can be located directly from its (plane, row, column) values, or from its surrogate Unicode value with a simple calculation. Specification, p. 14, ln. 23 – p. 15, ln. 2.

According to Figure 3, software 60 for implementing a browser 12 may begin by receiving an HTML web page in the plane, row or column (PRC) format as indicated in block 62. The plane is implicitly represented by the character set designator in the beginning of the text string and the row and column are represented by every two bytes in the text string.

Whenever a character set plane is changed, as determined at diamond 64, a new character set designator (CSG) is inserted to indicate a MOD change as indicated in block 66. For characters that are defined by the ISO-2022-CN and ISO-2022-CN-EXT, as determined at

diamond 68, table mapping may be used to map its 16-bit Unicode value as indicated in block 70. Thereafter, a rendered step is completed as indicated in block 72.

For a browser that supports 100,000 or more characters, those newly defined and encoded characters are mapped into the surrogate area as indicated in block 74. For Unicode characters in the surrogate areas, the surrogate value may be used to calculate the original values of plane, row and column. The browser is then directed to find its file location and the offset of a particular character's font bytes inside a font file as indicated in block 76. The glyph image is then generated as indicated in block 78 for either the ISO-2022-CN-EXT situation or the extended situation mapped to the surrogate areas. Specification, p. 16, ln. 5 – p. 17, ln. 14.

## **VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL**

Each of the following grounds of rejection are presented for review:

- (1) claims 1, 3-7, 11, 13-17, 21, 23-24 and 26-32 stand rejected under 35 U.S.C. § 103(a) over Powell, Taieb and Rojas
  - A. claims 1, 5, 6, 7, 11, 15-17, 21, 23-24 and 28 stand rejected under 35 U.S.C. § 103(a) over Powell, Taieb and Rojas
  - B. claims 3, 4, 13 and 14 stand rejected under 35 U.S.C. § 103(a) over Powell, Taieb and Rojas
  - C. claims 26, 29 and 32 stand rejected under 35 U.S.C. § 103(a) over Powell, Taieb and Rojas
  - D. claims 27 and 30 stand rejected under 35 U.S.C. § 103(a) over Powell, Taieb and Rojas
  - E. claim 31 stands rejected under 35 U.S.C. § 103(a) over Powell, Taieb and Rojas
- (2) claims 2, 12, 22 stand rejected under 35 U.S.C. § 103(a) over Powell, Taieb, Rojas and Lincke

## VII. ARGUMENT

### **(1) Claims 1, 3-7, 11, 13-17, 21, 23-24 and 26-32 Are Patentable Under 35 U.S.C. § 103(a) over Powell, Taieb and Rojas**

#### **A. Claims 1, 5, 6, 7, 11, 15-17, 21, 23-24 and 28 Are Patentable Under 35 U.S.C. § 103(a) over Powell, Taieb and Rojas**

Pending claims 1, 5, 6, 7, 11, 15-17, 21, 23-24 and 28 stand rejected under 35 U.S.C. §103(a) over U.S. Patent No. 5,157,905 (Powell) and U.S. Patent No. 6,718,519 (Taieb), and further in view of U.S. Patent No. 6,425,123 (Rojas). This rejection is improper and should be reversed. Claim 1 recites a method including receiving a file including characters, converting characters to a first code format having a double-byte length if the characters are of a first type, converting the characters to a second code format having a multiple double-byte length if the characters are of a second type, and displaying the characters using the two code formats.

As to claim 1, Powell nowhere teaches or suggests converting characters to a first code format having a double-byte length if the characters are of a first type and converting characters to a second code format having a multiple double-byte length if the characters are of a second type. This is so, as Powell does not perform “converting” as contended by the Examiner. Final Office Action (mailed October 21, 2005), pp. 2-3. Instead, Powell merely discloses that document representations are “characterized” or “mapped” to target values. Powell, col. 11, ln. 59 – col. 12, ln. 18. These characterizations are not conversions, as the original document representation still exists in its original form. In contrast the claimed characters are converted into a different format, not merely mapped.

Nor do any of the other cited references teach or suggest converting the characters to a first code format having a double-byte length if the characters are of a first type and converting the characters to a second code format having a multiple double-byte length if the characters are of a second type. In this regard, the Examiner concedes that Powell nowhere teaches or suggests this claimed conversion. Nor does Taieb, also conceded by the Examiner. Still further, Rojas, contended by the Examiner to teach this missing subject matter, also fails in this regard. Instead, Rojas merely teaches conversion of single-byte characters to double-byte equivalents.

In fact, Rojas nowhere teaches or suggests conversion of characters into a format having “a multiple double-byte length” whatsoever. Nor do any of the other references. In this regard,

the Examiner is utterly silent as to any teaching or suggestion in the prior art of conversion of characters into a code format having a multiple double-byte length. Final Office Action, pp. 2-4. Accordingly, the Examiner fails to set forth a *prima facie* case of obviousness with respect to the subject matter of claim 1. Thus, even accepting the combination of three references proposed by the Examiner, these three references fail to teach or suggest all limitations set forth in claim 1. Accordingly, claim 1 and its dependent claims are patentable over the proposed combination. MPEP §2143.03.

Further, there is no motivation to combine Powell and Taieb. In this regard, Powell is directed to statistical text analysis (Powell, col. 1, lns. 5-10). In contrast, Taieb is directed to displaying multiple language documents. The mapping of characters into 2D or 3D characterizations done in Powell is merely to determine a language in which the document was created. *E.g.*, Powell, col. 11, lns. 62-67. Powell teaches no other use for these 2D and 3D characterizations. Taieb, on the other hand, is directed to displaying multilingual texts. However, Taieb nowhere teaches or suggests use of conversion of characters to different code formats based on a type of the character.

Even more, the purported motivation to combine Rojas, Powell and Taieb is irrelevant to the subject matter of claim 1. In this regard, the Examiner contends it would have been obvious to combine Rojas with the other references “to include first code format having a double-byte length to create double-wide characters and the double-wide characters increase the spacing, i.e., field length, typically needed for translation of the text into a different language.” Final Office Action, p. 4. However, what is claimed is conversion of characters into code formats having given lengths, not creation of double-wide characters, nor increasing of spacing of such characters. As such, there is no teaching or suggestion in any of the three references to combine them in order to obtain the claimed subject matter. Accordingly, a *prima facie* case of obviousness has not been established. MPEP §2142; 2143.

Further, the combining of Powell and Taieb proposed by the Examiner would make no sense, and would not meet the recited subject matter of claim 1. That is, even if the 2D and 3D characterizations generated in Powell were displayed using the teaching of Taieb, no display of characters would result as recited by claim 1. Instead, a statistical 2D or 3D characterization would appear—not characters. Even more, the combination of Powell and Taieb with Rojas also fails to meet the subject matter of claim 1. That is, even if the characterizations generated in

Powell are displayed, the combination with Rojas still fails to teach or suggest conversion of characters to different code formats having different double-byte lengths depending upon a type of character in the received file.

Furthermore, the proposed combination would render the prior art unsatisfactory for its intended purpose. Accordingly, no suggestion or motivation can exist to make the proposed modification. MPEP §2143.01. In this regard, displaying the 2D and 3D characterizations of Powell based on the teaching in Taieb would frustrate the intended purpose of Powell, namely to perform statistical analysis to identify a language and character set. There is absolutely no reason to display the characterizations in the system of Powell. Even if displayed, the characterizations would not cause the display of the characters of the file as recited by the claims. Accordingly, for all these reasons the above-listed claims are patentable over the proposed combination and the rejection should be reversed.

**B. Claims 3, 4, 13 and 14 Are Patentable Under 35 U.S.C. § 103(a) Over Powell, Taieb and Rojas**

The rejection of claims 3-4 and 13-14 is further clearly erroneous, as these dependent claims depend from claims 2 and 12, which stand rejected under §103 over a combination of four references (*see* Heading (2), below). However, these claims 3-4 and 13-14 stand rejected under §103(a) only under the three references, Powell, Taieb and Rojas. Accordingly, the rejection of these claims is clearly erroneous as the Examiner fails to provide *prima facie* support in these three references (and in fact, the Examiner concedes that such subject matter is not present. Final Office Action, p. 6) for the subject matter of claims 2 and 12 from which these claims depend.

**C. Claims 26, 29 and 32 Are Patentable Under 35 U.S.C. § 103(a) Over Powell, Taieb and Rojas**

Claim 26 depends from claim 1 and further recites converting the characters to the first and second code formats before parsing a web page including the characters. The rejection of this claim under §103(a) is improper at least for the same reasons discussed above regarding claim 1. Furthermore, the rejection is improper as Taieb, contended to teach this subject matter nowhere does so. In this regard, the Examiner contends that Taieb teaches converting of characters before parsing a web page including such characters. For support, the Examiner refers to cols. 3-4 of Taieb. However Taieb nowhere teaches parsing a web page, and certainly does

not teach or suggest parsing a web page after converting characters. Instead, all that Taieb teaches is that characters of a document are parsed and tested against a character table bank to identify what character set can express the characters of a message. Taieb, col. 4, lns. 25-31. Nowhere, however, does this or any other portion of Taieb teach either parsing of a web page or doing so after conversion of characters. Instead, the system of Taieb is directed to text processing of messages. *Id.* For this further reason, claims 26, 29 and 32 are patentable, and the rejection should be reversed.

**D. Claims 27 and 30 Are Patentable Under 35 U.S.C. § 103(a) Over Powell, Taieb and Rojas**

Claim 27 is dependent from claim 1 and further recites that each converted character (i.e., of the first or second code format) is converted into an encoding and used to index into a font file to obtain a character. This claim stands rejected under 35 U.S.C. §103(a) over Powell, Taieb and Rojas. The rejection is improper at least for the same reasons discussed above regarding claim 1. In addition the rejection is improper as Powell, contended to meet this additional claimed subject matter nowhere does so. Specifically, the Examiner points to cols. 13-15 of Powell. Final Office Action, p. 5. However, all that this portion of Powell teaches is that script ranges of a Unicode character set map source values to target values. This nowhere, teaches conversion of a converted character (i.e., two conversion processes) and then indexing into a font file using the converted value (i.e., an encoding) to obtain a character. For this further reason, claims 27 and 30 are further patentable.

**E. Claim 31 Is Patentable Under 35 U.S.C. § 103(a) Over Powell, Taieb and Rojas**

Claim 31 depends from claim 30 and further recites using the encoding (i.e., a conversion of a converted character) to access a font file for characters of a particular type that are arranged in a row and column format. Claim 31 stands rejected under §103(a) over Powell, Taieb and Rojas. The rejection is improper at least for the same reasons discussed above regarding claims 1 and 30 both discussed above (*See* Subheadings A and D).

The rejection is further improper, as the Examiner contends support for the subject matter of claim 31 is met by FIG. 3 of Taieb. However, nowhere does either this figure or any other portion of Taieb teach or suggest a font file that is arranged in a row and column format. Instead, all that Taieb teaches is that a “big font” is capable of representing characters from

multiple scripts. Taieb, col. 4, lns. 56-67. Nowhere however, does Taieb teach that these big fonts are arranged in row and column formats. For this further reason, the rejection of claim 31 is improper and should be reversed.

**(2) Claims 2, 12, 22 Are Patentable Under 35 U.S.C. § 103(a) Over Powell, Taieb, Rojas and Lincke**

Pending claims 2, 12 and 22 stand rejected under 35 U.S.C. §103(a) over Powell, Taieb, and Rojas and in further view of U.S. Patent No. 6,397,259 (Lincke). The rejection is improper, at least for the same reasons discussed regarding claim 1 above. Further, as to claim 2, the Examiner concedes that none of Powell, Taieb or Rojas teaches or suggests receiving of a web page in a plane, row, and column format. Final Office Action, p. 6. Nor does Lincke, contended by the Examiner to meet this claim. *Id.* Instead, Lincke merely teaches that HTML documents include tags and attributes that are associated with text, tables and forms. Lincke, col. 21, ln. 65 – col. 22, ln. 10. However, nowhere does this or any other portion of Lincke teach or suggest that a web page is in a format including planes. Nothing in the Examiner's support from Lincke even remotely teaches or suggests use of such planes. Final Office Action, p. 6 (citing Lincke, col. 3, lns. 6-33 and col. 21, ln. 65-col. 22, ln. 8). Again, the Examiner fails to set forth any teaching, suggestion, or motivation for the claimed plane format and thus a *prima facie* case has not been set forth for these dependent claims. For these further reasons, claims 2, 12 and 22 are further patentable.

Furthermore, this combination of four references is nothing more than an improper hindsight reconstruction. That is, the Examiner has engaged in the hindsight-based obviousness analysis that has been widely and soundly disfavored by the Federal Circuit. In order to prevent a hindsight-based obviousness analysis, the Federal Circuit requires that “to establish obviousness based on a combination of the elements disclosed in the prior art, there must be some motivation, suggestion or teaching of the desirability of making the specific combination that was made by the applicant.” *In re Kotzab*, 55 U.S.P.Q.2d 1313, 1316-17 (Fed Cir. 2000). No such showing is present here. Instead, the alleged motivation for the combination set forth by the Examiner is that “it would have been obvious...to include a web page in a plane, row and column format in order to provide user friendly environment for web users.” Final Office Action, p. 6. This self-serving language is conclusory and fails to provide a legally sufficient

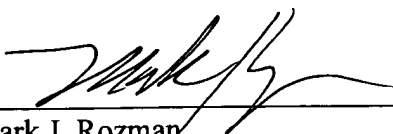


motivation to combine the references. *In re Lee*, 61 U.S.P.Q.2d 1430, 1435 (Fed. Cir. 2001). Thus the rejection should be reversed.

Applicant respectfully requests that each of the final rejections be reversed and that the claims subject to this Appeal be allowed to issue.

Respectfully submitted,

Date: March 15, 2006

  
\_\_\_\_\_  
Mark J. Rozman  
Registration No. 42,117  
TROP, PRUNER & HU, P.C.  
8554 Katy Freeway, Suite 100  
Houston, Texas 77024-1805  
(512) 418-9944 [Phone]  
(713) 468-8883 [Fax]  
Customer No.: 21906

### **XIII. CLAIMS APPENDIX**

The claims on appeal are:

1. A method comprising:  
receiving a file including characters;  
converting the characters of said file to a first code format having a double-byte length if the characters are of a first type;  
converting the characters of said file to a second code format having a multiple double-byte length if said characters are of a second type; and  
displaying the characters of the file using the first code format or the second code format.
2. The method of claim 1 including receiving a web page in a plane, row and column format.
3. The method of claim 2 including checking to determine whether a character set plane is changed.
4. The method of claim 3 wherein if the character set plane is changed, inserting a previously presented character set designator.
5. The method of claim 1 including determining whether the characters in the file are defined according to the first code format.
6. The method of claim 5 wherein if said characters are coded according to said first code format, table mapping Unicode values to said first code format.
7. The method of claim 5 wherein if said first code format is not utilized, using a surrogate area of Unicode.
11. An article comprising a medium storing instructions that enable a processor-based system to:  
receive a file including characters;

convert the characters of the file to a first code format having a double-byte length if the characters are of a first type;

convert the characters of said file to a second code format having a multiple double-byte length if said characters are of a second type; and

display the characters of the file using the first code format or the second code format.

12. The article of claim 11 further storing instructions that enable the processor-based system to receive a web page in a plane, row and column format.

13. The article of claim 12 further storing instructions that enable the processor-based system to determine when a character set plane is changed.

14. The article of claim 13 further storing instructions that enable the processor-based system to insert a previously presented character set designator if the character set plane is changed.

15. The article of claim 11 further storing instructions that enable the processor-based system to determine whether the characters in the file are defined according to the first code format.

16. The article of claim 15 further storing instructions that enable the processor-based system to map Unicode values to said first code format if said characters are coded according to said first code format.

17. The article of claim 15 further storing instructions that enable the processor-based system to use a surrogate area of Unicode if said first code format is not utilized.

21. A system comprising:

a processor; and

a storage coupled to the processor, the storage storing a browser that is able to receive a file including characters, convert the characters of the file to a first code format having a double-byte length if the characters are of a first type, convert said file to a second code format having a multiple double-byte length if said characters are of a second type, and display the characters of the file using the first code format or the second code format.

22. The system of claim 21 wherein the storage stores instructions that enable the processor to receive a web page in a plane, row and column format.

23. The system of claim 21 wherein said storage stores instructions that enable the processor to map Unicode values to the first code format if the characters are coded according to the first code format.

24. The system of claim 23 wherein said storage stores instructions that enable the processor to use a surrogate area of Unicode if the first code format is not utilized.

26. The method of claim 1, further comprising converting the characters to the first code format or the second code format before parsing a web page including the characters.

27. The method of claim 1, wherein displaying the characters comprises converting each converted character into an encoding and indexing into a font file using the encoding to obtain the character.

28. The method of claim 1, wherein the second code format accommodates at least 100,000 characters.

29. The article of claim 11, further storing instructions that enable the processor-based system to convert the characters to the first code format or the second code format before parsing a web page including the characters.

30. The article of claim 11, wherein further storing instructions that enable the processor-based system to convert each converted character into an encoding and index into a font file using the encoding to obtain the character.

31. The article of claim 30, further storing instructions that if executed enable the processor-based system to use the encoding to access the font file for the characters of the second type arranged in a row and column format.

32. The system of claim 21, wherein the processor is to convert the characters to the first code format or the second code format before parsing a web page including the characters.